

5 Conclusão

A busca pela melhoria no uso dos recursos computacionais tem causado um crescente interesse em infraestruturas computacionais compartilhadas, tais como ambientes de grades computacionais e computação em nuvem. Nessas infraestruturas é importante que o desempenho previsto para uma aplicação não seja influenciado pela execução de cargas de trabalho sendo executadas na mesma máquina. Uma forma de atingir esse objetivo é garantir o acesso das aplicações aos recursos através de reservas.

Este trabalho apresentou um estudo sobre a viabilidade e a eficácia de reserva de recursos providas no nível do usuário. Motivado pela necessidade de controladores de recursos em infraestruturas computacionais de uso compartilhado, este trabalho vai ao encontro de uma tendência de disponibilização de gerenciamento de recursos no espaço do usuário. Provas disso são o ressurgimento do uso da técnica de virtualização e as recentes melhorias para o controle de recursos introduzidas no núcleo do sistema operacional Linux.

Dado o objetivo desta tese de investigar mecanismos que provessem a garantia e a limitação do uso de recursos, foi preciso estudar as iniciativas já existentes na literatura e relacionadas a esse tema. O estudo realizado mostrou que muitos trabalhos, com diferentes técnicas para a provisão de reserva, existem, mas, por diferentes razões, apresentam algumas limitações. Exemplos dessas limitações consistem na dificuldade de se estender as funcionalidades das ferramentas e na ausência de reservas concomitantes de processamento e de largura de banda de disco. Essas limitações motivaram o projeto de uma nova ferramenta que tratasse dos recursos de processamento e de largura de banda de disco. Tal ferramenta, denominada ReservationSuite, foi sendo incrementada com novas funcionalidades que atendem a requisitos encontrados em cenários computacionais mais atuais. Foram inseridas, por exemplo, a funcionalidade de reserva concomitante de recursos, para o atendimento a aplicações com uso intensivo de múltiplos recursos, e a funcionalidade de flexibilização das políticas de escalonamento, a fim de servir a aplicações com objetivos de desempenho diversificados. Além disso, para prover um isolamento de desempenho de maior granularidade, a ferramenta, quando utilizada em

arquiteturas multiprocessadas, também apresenta um mecanismo de reserva de subconjuntos de processadores.

O desenvolvimento de uma nova ferramenta nos ajudou a identificar as dificuldades de se implementar mecanismos de reserva no nível do usuário. Notamos que essas dificuldades são provenientes, principalmente, da perda de controle de recursos quando a técnica é comparada com trabalhos que alteram o núcleo do sistema operacional. Ao considerar o controle no acesso ao disco, por exemplo, essa dificuldade torna-se bastante evidente. Pelo disco rígido ser considerado uma mídia de acesso lento, os sistemas operacionais implementam otimizações para acelerar a taxa de vazão desse dispositivo. Essas otimizações dificultam a determinação dos pontos de saturação dos discos e os consequentes tempos de resposta que podem ser obtidos, métricas essenciais para a provisão de qualidade de serviço. No entanto, experimentos mostraram que, se a ferramenta de reserva for utilizada com um rígido controle de admissão, o qual não aceite novos pedidos de reserva quando o ponto de saturação dos recursos for atingido, as reservas são garantidas. Além disso, acreditamos que as dificuldades de se controlar o acesso à mídia de armazenamento tendem a ser reduzidas à medida que dispositivos de armazenamento de estado sólido (SSD) se tornarem populares. Nesse tipo de mídia, os tempos de acesso aos dados são reduzidos, pois não existem partes móveis na unidade de armazenamento.

A dificuldade causada pelo controle limitado do uso de recursos também se reflete no tipo de aplicações candidatas a serem gerenciadas pelas ferramentas implementadas no nível do usuário. Tais aplicações não podem apresentar fortes características de tempo real como é o caso de sistemas de controle de tráfego aéreo e outros sistemas críticos, onde atrasos no tempo de resposta das aplicações não podem acontecer. No entanto, aplicações onde variações de desempenho são toleráveis, assim como aplicações que precisam ter o seu consumo de recursos limitado, são muito beneficiadas pelo uso dessas ferramentas. Aplicações desse tipo são encontradas, por exemplo, em infraestruturas de grades computacionais.

Experimentos mostraram também que a sobrecarga de trabalho gerada pelo controle de recursos é mínima ao tratar reservas de processamento, mas pode ser significativa ao tratar reservas de E/S de dados. Porém, acreditamos que o desempenho da ferramenta pode ser melhorado e que os benefícios trazidos pelo controle de E/S podem superar essa desvantagem, principalmente quando aplicações com acesso esporádico ao disco são manipuladas.

A fim de determinar a eficácia de múltiplas reservas feitas a uma única aplicação pelo ReservationSuite, era preciso realizar experimentos com uma aplicação que fizesse uso intensivo dos recursos controlados pela nossa fer-

ramenta. A aplicação escolhida foi o TPC-W, uma aplicação complexa, de multicamadas e que, por simular o funcionamento de um sítio de comércio eletrônico atualmente muito acessado, faz uso intensivo dos recursos de processamento, disco e rede. Uma outra importância do uso do TPC-W vem do fato desse tipo de aplicação ser muito utilizada em ambientes de computação sob-demanda e com requisitos de qualidade de serviço. Assim, nos testes envolvendo o TPC-W era preciso que relações entre o número de requisições atendidas pela aplicação e as reservas de processamento e acesso ao disco fossem estabelecidas, realizando uma decomposição de QoS.

Para esses experimentos, um método de predição de desempenho simples, baseado em uma técnica de regressão por quadrados mínimos, foi apresentada. Esse método se mostrou bastante eficiente na tarefa de predição. Porém, notou-se que a eficácia da metodologia utilizada pode ser bastante dependente da medida utilizada como variável preditora da regressão. Em aplicações com grande variação de carga de trabalho, por exemplo, é preciso utilizar medidas que expressem tal variação, visto que, nesses casos, o uso da média de consumo tende a ser ineficiente para a previsão de QoS. Acreditamos, porém, que determinar a medida utilizada na predição consiste em um trabalho externo à nossa investigação, apesar de ser tão importante quanto.

Conclui-se então que, somente com informações providas por um sistema operacional de propósito geral atual e sem causar demasiada sobrecarga no ambiente de execução, é possível implementar mecanismos capazes de limitar e garantir a execução de aplicações em cenários de Computação Compartilhada, bastando que as ferramentas desenvolvidas sejam utilizadas juntamente com mecanismos de controle de admissão.

Apesar desta tese ser focada na garantia de desempenho, acreditamos que a ferramenta de reserva apresentada pode ser utilizada para diversos outros objetivos. A manipulação da porcentagem de recursos atribuída a uma aplicação, por exemplo, pode ser capaz de controlar o nível de energia das baterias de dispositivos móveis e a redução do aquecimento produzido por computadores, assim como sugerem alguns trabalhos envolvendo escalonadores de tempo real [53, 54]. Além disso, aliada à reserva de processadores, essa manipulação possibilitaria a construção de ambientes de testes de escalabilidade com variadas capacidades computacionais. Por fim, a flexibilidade para definição de regras de escalonamento presente no ReservationSuite pode facilitar o estudo de novas políticas de compartilhamento de recursos.

Dois casos de uso do ReservationSuite devem ser destacados. No primeiro, parte do ReservationSuite, mais precisamente a parte relativa à reserva de processamento (CPUReserve), foi inserida no gerenciamento de recursos do

InteGrade, um *middleware* para grades computacionais oportunistas que permite a execução de aplicações paralelas altamente acopladas [87, 61]. Originalmente, o InteGrade executava aplicações da grade com a prioridade mínima permitida para processos comuns. Esse fato fazia com que os processos do usuário doador não fossem prejudicados no acesso ao processamento, porém, essa prática não evitava que os processos da grade consumissem todo o processamento disponível na máquina local em momentos em que a máquina encontrava-se em estado ocioso. Tal fato dificultava a adesão de novos computadores na grade [26]. Assim, o CPUReserve foi integrado ao InteGrade para que pudesse limitar o uso de processamento dos processos da grade. Como havia uma necessidade de que a ferramenta de gerenciamento utilizada no InteGrade não precisasse de privilégios para ser executada, o CPUReserve foi alterado para não mais manipular a prioridade das aplicações além do que é, por padrão, permitido a processos comuns do Linux. Com essa alteração, o CPUReserve é capaz apenas de limitar o uso de processamento, e não mais de garantir esse uso no caso de haver outras cargas de trabalho competindo pela CPU.

O segundo cenário consiste no uso do ReservationSuite como ferramenta auxiliar de um gerador de carga. A ideia é que utilizando o mecanismo de limitação de uso de processamento, seja possível determinar as características de consumo que cargas de trabalho sintéticas devem assumir. No caso do gerador descrito, essas características são importantes, por exemplo, para simular ambientes e analisar o comportamento de sistemas de gerenciamento [27].

5.1 Trabalhos Futuros

Durante a elaboração desta tese foi possível identificar alguns temas interessantes para investigação futura. Entre os temas relacionados à provisão de reservas estão:

- A adição de novas funcionalidades ao ReservationSuite: algumas funcionalidades podem ser inseridas no ReservationSuite, sendo a reserva de memória a mais importante delas. Com a inclusão desse novo tipo de reserva, seria possível um maior controle do uso da memória e também da utilização das larguras de banda de disco e de rede. Uma outra funcionalidade consiste na implementação de uma API para reservas de menor granularidade. Com essa API, as aplicações poderiam, em seus próprios códigos, indicar quais partes de sua implementação precisam de maior garantia de desempenho;

- O estudo de novas políticas: nenhum estudo de políticas de escalonamento envolvendo todos os recursos gerenciados pelo ReservationSuite foi conduzido. Com a ferramenta proposta, muitas políticas baseadas em processamento e E/S de dados podem ser criadas ou reproduzidas para serem estudadas;
- O estudo de novas aplicações da ferramenta: espera-se que o ReservationSuite seja utilizado em outros tipos de cenários, em especial, em *clusters* onde o superaquecimento das máquinas é um problema. Com o controle de uso de recursos, gostaríamos de investigar uma possível relação entre os consumos de processamento e de largura de banda de disco e a produção de calor. Gostaríamos de verificar se há similaridades entre o uso do controle de uso de recursos com o objetivo de economizar energia em um dispositivo móvel e o uso desse controle com o objetivo de reduzir o aquecimento de computadores. A ideia é que essa investigação ajude a evitar o desperdício de energia e o mau funcionamento de máquinas por superaquecimento;
- A integração com *middlewares* de gerenciamento: existem dois projetos de *middleware* desenvolvidos por nosso grupo de pesquisa [88] que se beneficiariam muito com a inclusão das funcionalidades providas pelo ReservationSuite. O primeiro deles, é o CSBase, um *framework* para gerenciamento de recursos e execução de algoritmos em ambientes computacionais distribuídos e heterogêneos [89]. A infraestrutura de execução do CSBase é compartilhada por diferentes usuários sem que mecanismos eficientes de isolamento de desempenho estejam presentes. Esse fato, que causa problemas de desempenho e, muitas vezes, subutilização das máquinas executoras, pode ser contornado com o uso do ReservationSuite. O segundo projeto consiste no SCS (*Software Component System*), um sistema de componentes de *software* cujo objetivo é prover uma infraestrutura de *middleware* apropriada à implantação de componentes [90]. Ao considerarmos aplicações implementadas sob o paradigma de componentes, é natural imaginar o uso do ReservationSuite para prover garantias de desempenho às unidades de *software* hospedadas nos contêineres de execução do SCS.

Outros trabalhos aos quais desejamos dar prosseguimento envolvem a implementação de outros módulos relativos à arquitetura de gerenciamento de recursos descrita no início do Capítulo 3 e melhorias no método utilizado para a decomposição de QoS.

Entre os módulos da arquitetura de gerenciamento, temos especial interesse na implementação do Controlador de Admissão, pois a reserva de recursos

é intimamente ligada a ele. Esse módulo poderia ser responsável por estimar o ponto de saturação de dispositivos de E/S, negociar taxas de qualidade de serviço quando não houver recursos suficientes disponíveis e informar quando adaptações de reservas podem ser realizadas.

Acreditamos que o método proposto para a decomposição de QoS possa ser melhorado com o uso de técnicas mais avançadas para a predição de desempenho. Particularmente, acreditamos que é possível melhorar a precisão das regressões ao considerarmos métricas de desempenho mais específicas que o número de interações *web* utilizada em nossos experimentos. Além disso, também gostaríamos de investigar mais a fundo o impacto que a variação no uso de recursos causa no uso de médias como variáveis de entrada das predições. Entender essa relação, pode ser fundamental na melhoria de nosso método.